In the United States Patent and Trademark Office
Board of Patent Appeals and Interferences

Appeal Brief

In re the Application of:

METHOD, SYSTEM, AND PROGRAM FOR PREPROCESSING A DOCUMENT TO
RENDER ON AN OUTPUT DEVICE

Steven Mark Gebert, David George Greenwood, Reinhard Heinrich Hohensee, Harry
Reese Lewis, Jr., Dwight Ross Palmer, Arthur Ray Roberts, and David Earl Stone

Serial No. 09/782,850
Filed: February 14, 2001
Attorney Docket No. BLD920000048US1

Submitted by:

Konrad, Raynes & Victor LLP
315 So. Beverly Dr., Ste. 210
Beverly Hills CA 90212
(310) 556-7983
(310) 556-7984 (fax)

TABLE OF CONTENTS

I.   Real Party in Interest

        The entire right, title and interest in this patent application is assigned to real party

in interest International Business Machines Corporation.

II.   Related Appeals, Interferences, and Judicial Proceedings

        Appellant, Appellant's legal representative, and Assignee are not aware of any

other prior or pending appeals, interferences, and judicial proceedings which may be

related to, directly affect or be directly affected by or have a bearing on the Board's

decision in the pending appeal.

III.   Status of the Claims

        Claims 1, 2, 4-11, 14-16, 18-25, 28-30, 32-39, and 42-48 are pending and have

been rejected in view of prior art.

IV.   Status of Amendments

        On May 2, 2006, Applicants submitted an amendment under 37 CFR 41.33 to

change the dependency of claim 19 to intervening claim 16 to place this claim in better

form for consideration on Appeal.   As of the filing of this Appeal Brief, the Examiner

has not ruled on this Amendment.

V.   Summary of the Claimed Subject Matter

        Independent claims 1, 15, and 29 include a preamble reciting processing a source

document in a structured document format including elements providing source content

to render, wherein the source content comprises code that is rasterized into output. With respect to the preamble, the Specification mentions that the preprocessor 4 (FIG. 1) receives an XML source document 14. (Specification, pg. 10, lines 10-14). An XML document is a structured document having elements and the content may be rasterized and printed. (Id. at pg. 1, lines10-26, pg. 13, lines 3-5).

These claims recite receiving the source document including source content in a presentation language. With respect to this requirement, block 100 of FIG. 3 discloses receiving an XML. (Id. at pg. 10, lines 10-14). The XML source document may be in XML or expressed in a different presentation language other than XML or XSL formatting objects, such as DHTML, SGML, etc.. (Id. at pg. 14, lines 13-25).

The claims further recite receiving a layout data structure separate from the source document, providing formatting properties specifying a layout and format of the content output, wherein the layout data structure does not include source content. With respect to this requirement, block 102 of FIG. 3 discloses receiving an XSL stylesheet. (Id. at pg. 10, lines 12-13). An XSL stylesheet includes a transformation language, formatting language, and formatting properties. The term "formatting properties" describes any information used to express the layout and presentation of the content, such as page layout, fonts, page size, element size, font color, etc. (Id. at pg. 1, line 26; pg. 2, lines 19-20; pg. 3, lines 2-7; pg. 7, lines 1-5).

The claims further recite processing the source document and the layout data structure to determine formatting properties, including page divisions, for the content in the source document. Block 102 in FIG. 3 recites that the XSL processor 12 transforms the XML code in the source document 16 into the result document 20 having XML

source content and XSL formatting objects (XSL-FO) according to the page layout and formatting properties specified in the XSL stylesheet in a manner known in the art. (Id. at p. 10, lines 12-17)

The claims further recite generating multiple page objects. Block 106 of FIG. 3 recites calling a page separator program to generate page objects. (Id. at pg. 10, lines 17-19).

The claims recite that each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page. The Specification discloses that the page objects 22a…n includes the content and layout information for each page in the result document, and the content in the page object 22a…2n includes the coding in the language of the original source document 16. (Id. at pg. 7, line 23 to pg. 8, line 6).

The claims further recite that at least one page object has multiple content elements and that the content elements include content to place on the pages. Blocks 110-130 of FIG. 3 disclose how the page separator adds content elements to a page object while there is room in the page for content elements. (pg. 10, line 23 to pg. 11, line 21).

The claims further recite transmitting the page objects to a rasterizer to transform into renderable information capable of being generated by an output device. The page objects 22a…2n are sent to a rasterizer 26 to transform the page objects 22a…h into printer ready raster data that can be rendered by the printer mechanisms 28.

Independent claim 15 additionally recites an output device and recites the functional language as "means" limitations. The Specification discloses a printer 8 output device. (FIG. 1). With respect to the "means" limitations, the corresponding

structure, materials and acts disclosed in the Specification corresponding to the claimed

means functions of receiving the source document, receiving the layout data structure,

processing the source document and the layout data structure, generating multiple page

objects, and transmitting the page objects comprises a XML preprocessor 4 shown in

FIG. 1, which is provided the XML source document and XSL stylesheet. As shown in

FIG. 1, the XSL processor 12 component is the component in the preprocessor 4 that

receives the source document 16 and the XSL stylesheet 18 as 1 and transforms the

source document 16 to a result document 20. (Id. at pg. 6, line 1 to pg. 7, line 5) The

page separator 14 is the means or component of the XML preprocessor 4 that generates

the multiple page objects that are transmitted to the printer 8. (Id. at pg. 7, line 21 to pg.

9, line 21) , line 5 pg. 10, lines 10-14).

Claim 16 depends from claim 15 and additionally recites a means for

transforming the source document and source content into a result document in a second

presentation language. The corresponding structure, materials and acts disclosed in the

Specification corresponding to the claimed means functions for transforming the source

document into a result document comprises the XML preprocessor 4 shown in FIG. 1,

which is provided the XML source document and XSL stylesheet. As shown in FIG. 1,

the XSL processor 12 component is the component in the preprocessor 4 that receives the

source document 16 and the XSL stylesheet 18 and transforms the source document 16 to

a result document 20. (Id. at pg. 6, line 1 to pg. 7, line 5) The source document

presentation language is disclosed as XML and the second presentation language of the

result document 20 includes XSL formatting objects. (Id. at p. 6, lines 21-26)

VI.   Grounds of Rejection to Be Reviewed on Appeal

A concise statement listing each ground of rejection presented for review is as

follows:

A.   Whether claims 1, 2, 4-11, 14-16, 18-25, 28-30, 32-39, and 42-48 are

unpatentable as obvious (35 U.S.C. §103(a)) over Adler ("Extensible Stylesheet

Language (XSL), Version 1.0, published on Oct. 18, 2000) in view of Saito (U.S. Patent

No. 5,323,312). .

B.   Whether claims 5, 7, 19, 21, 21, 33, and 35 are unpatentable as obvious

(35 U.S.C. §103) over Adler in view of Saito and further in view of Barry (U.S. Patent

No. 6,606,165).

C.   Whether claims 6, 20, and 34 are unpatentable as obvious (35 U.S.C.

§103) over Adler in view of Saito, further in view of Barry, and further in view of Sall

(FOP: Formatting Object to PDF Translator (James Tauber) by Ken Sall, May 24, 1999).

VII.   Argument

A.   Rejection Under 35 U.S.C. §103(a) Over Adler in View of Saito

1.   Claims 1, 4, 9-11, 14, 15, 18, 23-25, 28, 29, 32, 37-39,

42, 43, 45, and 47

Claims 1, 15, and 29 concern processing a source document in a structured

document format including elements providing source content to render, wherein the

source content comprises code that is rasterized into output, and require: receiving the

source document including source content in a presentation language; receiving a layout

data structure separate from the source document, providing formatting properties

specifying a layout and format of the content output, wherein the layout data structure does not include source content; processing the source document and the layout data structure to determine formatting properties, including page divisions, for the content in the source document; generating multiple page objects, wherein each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page, wherein at least one page object has multiple content elements, and wherein the content elements include content to place on the pages; and transmitting the page objects to a rasterizer to transform into renderable information capable of being generated by an output device.

Adler discusses an XSL stylesheet processor that accepts a document or data in XML and an XSL stylesheet and produces the presentation of that XML source content as intended by the designer of the stylesheet. There are two parts of the presentation process, first constructing a result tree and then interpreting the result tree to produce formatted results suitable for presentation on display, paper or other media. (Adler, pg. 17). The result tree includes formatting semantics comprising formatting objects that are elements of the tree that denote abstractions such as page, paragraph, table, etc. (Adler, pg. 18).

The Examiner cited pgs. 20-21 and 25-27 of Adler as disclosing the claim requirement of generating the page objects. (Final Office Action, pgs. 3-4) Applicants traverse with respect to the claim requirement of generating multiple page objects, wherein each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page, wherein at

least one page object has multiple content elements, and wherein the content elements include content to place on the pages.

The cited pages 20-21 of Adler discuss how formatting objects are generated from an XML document and an XSL stylesheet. Each formatting object represents a kind of formatting behavior, such as the formatting of a paragraph, color, space before, etc. Formatting consists of the generating of a tree of geometric areas positioned on a sequence of one or more pages. According to Adler, each geometric area has a position on the page, a specification of what to display, and borders.

Although the cited pg. 20 discusses geometric areas on pages, nowhere does the cited pg. 20 anywhere teach that the formatting objects comprise page objects, each for one page, including source content in the presentation language used in the source document. Moreover, nowhere does the cited pg. 20 anywhere teach or suggest generating page objects and that at least one page object has multiple content elements including content to place on the pages. Instead, the cited pg. 20 discusses how a tree defines geometric areas positioned on a sequence of on one or more pages.

Applicants further submit that the cited pg. 20 teaches away from the claim requirement because the cited pg. 20 discusses generating an area tree of geometric areas, not page objects, where each page object has source content in the presentation language used in the source document and determined formatting properties for one page.

The cited pg. 21 discusses the steps in formatting to "objectify" the element and attribute tree obtained via an XSLT transformation by turning the elements in the tree into formatting object nodes, the attributes into property specifications, the characters replaced with fo:character nodes, etc. Nowhere does the cited pg. 21 teach or suggest the

claim requirements of generating multiple page objects, wherein each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page, and that at least one page object has multiple content elements.

Moreover, the cited page 21 teaches away from the claim requirement because the result of the formatting step discussed on page 21 is to generate a formatting object tree, not multiple page objects where each page object has source content and formatting properties for one page as claimed.

Page 23 of Adler further emphasizes the differences with the claim requirement of multiple page object. Page 23 mentions that the last part of formatting describes the generation of a tree of geometric areas positioned on a sequence of one or more pages. Thus, Adler produces a tree that has geometric areas of data to place on a page. This is different from and does not teach multiple page objects where each page object has source content and formatting properties for one page.

The cited pg. 25 discusses the goal of XSL to provide designers control over features. The cited pg. 27 discusses formatting objects that describe both the layout structure of a page or frame and the rules by which the XML source content is placed in the containers - formatting objects. The formatting object allows one to define independently filled regions for the body. Page sequences specify the order in which page masters will be used. The page sequence also specifies how styled content is to fill those pages.

Although the cited pg. 27 discusses how page sequences specify how styled content is used to fill pages, nowhere does the cited pg. 27 teach generating page objects

including source content in the presentation language used in the source document and

the determined formatting properties for one page. Further, nowhere does the cited pgs.

25-27 teach or suggest the claim requirement that at least one page object has multiple

content elements including content to place on the pages.

Instead, the cited page 27 discusses XSL formatting objects providing the

structure for a page. However, these formatting objects that provide page layout

information are organized in a formatting object tree. (Adler, pg. 22) Applicants submit

that a tree of formatting objects having page layout information does not teach the claim

requirement of multiple page objects where each page object has source content and

formatting properties for one page. Instead, the cited formatting objects of Adler provide

layout and other information, not objects such that each object has content and layout

information for one page as claimed.

The Examiner found that "Adler discloses that the page objects include the

content that is to be placed on the pages" (Final Office Action, pg. 4) Applicants

traverse because as discussed the formatting objects of Adler are elements in a tree that

provide formatting information, not objects where each object has content and layout

information for one page as claimed.

In the Response to Arguments, the Examiner cited pg. 18 and Final Paragraph and

Figure of Adler as disclosing a result tree consisting of formatting objects. The

Examiner found that because Adler mentions that the classes of formatting objects denote

abstractions such as page, the formatting objects in the result tree of Adler disclose the

claimed multiple page objects. (Final Office Action, pgs, 8-9) Applicants traverse.

The Examiner has not cited any part of Adler or other art that teaches or suggests the claim requirement that each of the page objects include source content in the presentation language of the source document and formatting properties for one page. Instead, the cited pg. 18 of Adler mentions that the formatting object classes denote typographic abstractions, such as page, paragraph, table, so forth. This statement in Adler which the Examiner cites does not teach that each of the page objects includes for one page determined formatting properties and content in the presentation language used in the source document. Moreover, there is no disclosure in the cited Adler that a formatting object of the page class, noted by the Examiner, comprises a page object having formatting properties and source content in the presentation language of the source document for one page as claimed.

The Examiner further found that because pg. 18 of Adler mentions that the formatting objects are represented as an XML element, this discloses the claim requirement that the page object (which the Examiner likens to the formatting objects) includes source content in the presentation language used in the source document. (Final Office Action, pg. 9) Applicants traverse this finding because the cited pg. 18 of Adler mentions that a "formatting object can be represented as an XML element, with the properties represented by a set of XML attribute – value pairs. The content of the formatting object is the content of the XML element." Thus, the cited pg. 18 mentions that the formatting object has XML elements, not that the formatting object includes source content from the source document in the presentation language used in the source document as claimed.

Moreover, the distinction between the claimed page objects and the cited tree nodes comprising formatting objects is further emphasized by the claim requirement of transmitting the page objects to a rasterizer to transform into renderable information. The Examiner has not shown where Adler teaches or suggests that the formatting objects or nodes in the result tree each comprise the information for one page that are then rasterized.

The Examiner recognized that Adler does not disclose that the content consists of multiple content elements, but cited col. 1, lines 31-57 of Saito as disclosing that it was well known in the art that a structured document could consists of two parts a layout structure and logical structure (source content). (Final Office Action, pg. 4). Applicants traverse.

The cited col. 1 of Saito mentions that a document has a layout structure and a logical structure , where the content of a document is linked to logical objects positioned at the lowest level of a specific logical structure. This structure enables automatic layout processing. The cited col. 1 discusses a process in which if a certain document is not laid out in a subordinate structure of a page object, a new page object is generated and the overflown document is laid into the new page object.

Although the cited Saito discusses page objects, nowhere does the cited Saito teach or suggest the claim requirement that each page object has source content in the presentation language used in the source document and determined formatting properties for one page. Further, nowhere does the cited Saito teach that the page object has multiple content elements, each having content to place on the pages. Instead, the cited Saito discusses laying document content onto a page object.

FIG. 7 of Saito shows how a document is laid out in a structure of paragraphs, and that the content of the paragraphs of a section is laid out on a first page, and if the content is to long the content is laid out on a second page. (Saito, col. 1, line 58 to col. 2, line 2). Although Saito discusses how content is laid out into pages, there is still no teaching or suggestion that each page object includes source content in the presentation language used in the source document and determined formatting properties for one page, as well as multiple content elements.

Further, according to the Specification, a "presentation language provides constructs to use to specify the formatting properties to present the content. The document described with the presentation language can be interchanged with different application programs and output devices such that the presentation document produces the same document content in the same format on different printers or display devices dependent, however, on the capabilities of each of the printers or display devices." (Specification, pg. 8, lines 9-14).

The cited Saito discusses putting paragraph content onto a page, not source content in a presentation language used in the source document, as the term "presentation language" is defined in the specification.

Because the cited combination of art does not teach or suggest all the claim requirements, alone or in combination, Applicants submit that the rejection of claims 1, 15, and 29 should be withdrawn.

Applicants submit that the rejections with respect to dependent claims 4, 9-11, 14, 18, 23-25, 28, 32, 37-39 42, 43, 45, and 47 should be withdrawn because these claims

depend from claims 1, 15, and 29, which are patentable over the cited art for the reasons discussed above.

2. Claims 2, 16, and 30

Claims 2, 16, and 30 depend from claims 1, 15, and 30 and further require that the source document includes statements in a first presentation language and transforming the source document and source content therein into a result document in a second presentation language, wherein the result document includes the source content and the formatting properties provided by the layout data structure, wherein the formatting properties indicate page divisions of the content, and wherein the multiple page objects are generated from the result document.

The Examiner cited pg. 17-18 of Adler as disclosing different formats in the XML source document and the XSL-FO result document. (Final Office Action, pg. 4) Applicants submit that although the source and result documents are in different formats, the Examiner has not cited any part of Adler that teaches generating from the result document page objects that include source content in the first presentation language used in the source document. Instead, the cited Adler discusses how the result and source documents are in different presentation languages.

The Examiner then cited pgs. 20-21 and 25-27 of Adler as teaching page objects generated by filling XML content into containers. (Final Office Action, pg. 4) The cited pg. 27 of Adler mentions that the XSL formatting objects provide rules by which XML source content is placed in "containers". Although this cited pg. 27 mentions "containers" of XML source content, the Examiner has not cited any part of Adler that

teaches or suggests that each of these cited "containers" provide content and formatting properties for one page as claimed.   Instead, the cited Adler just mentions that the containers have XML source content.

In the Response to Arguments, the Examiner found, as discussed above, that the formatting objects of Adler include source content in the presentation language of the source document. (Final Office Action, pgs. 8-9) Applicants traverse these findings for the reasons discussed above with respect to claims 1, 15, and 29.

Further, the Examiner has not cited any part of Adler or Saito that teaches or suggests generating the page objects from the result document. Page 18 of Adler shows a result tree being formatted and sent to a printer.   According to Adler, the result tree has "objects primarily in the 'formatting object' namespace". (Adler, pg. 18) The Examiner has not cited any part of Adler that teaches or suggests that the formatting objects, which the Examiner likens to the claimed page objects and which are nodes of the result tree, are generated from the result tree as claimed. Instead, in Adler, the formatting objects are nodes of the result tree and, thus, cannot be generated from the result tree, i.e., the examiner has not cited where Adler mentions the formatting objects are generated from a result tree having nodes of formatting objects.

Because the cited combination of art does not teach or suggest all the claim requirements, alone or in combination, Applicants submit that the rejection of claims 2, 16, and 30 should be withdrawn.

3. Claims 8, 22, and 36

Claims 8, 22, and 36 depend from claims 2, 16, and 30 and further recite the page objects include content and formatting properties in the second presentation language, which is the language of the result document.

The Examiner cited pgs. 20-21 and 25-27 of Adler as teaching the additional requirements of these claims. (Final Office Action, pg. 5)

As discussed, the cited pgs. 20-21 discusses formatting objects and the cited pgs. 25-27 also discusses formatting objects and page sequences. Nowhere does the cited Adler anywhere teach that page objects generated from the result document include source content in the first presentation language of the source document as well as content and formatting properties in the second presentation language used in the result document. The Examiner has not cited any part of Adler that teaches or suggests generating page objects from the result document having content and formatting properties in the first and second presentation languages. For instance, pg. 18 of Adler shows a result tree going to an output device via an XSL formatter, but nowhere shows generating page objects including content and formatting properties from the result document in presentation languages as claimed.

Because the cited combination of art does not teach or suggest all the claim requirements, alone or in combination, Applicants submit that the rejection of claims 8, 22, and 36 should be withdrawn.

4. Claims 44, 46, and 48

Claims 44, 46, and 48 depend from claims 43, 45, and 47 and further require that page sequence elements include content elements and accessing page sequence elements according to an ordering of the page sequence elements, wherein the content elements within the accessed page sequence elements are added to page objects.

The Examiner cited col. 1, lines 31-57 of Saito as teaching these claim requirements. (Final Office Action, pgs. 5-6) Applicants traverse.

The cited col. 1 of Saito mentions that a document has a layout structure and a logical structure , where the content of a document is linked to logical objects positioned at the lowest level of a specific logical structure. This structure enables automatic layout processing. The cited col. 1 discusses a process in which if a certain document is not laid out in a subordinate structure of a page object, a new page object is generated and the overflown document is laid into the new page object.

Although the cited Saito discusses laying out document content in page objects, nowhere is there any teaching of page sequence elements including content elements, such that the page sequence elements are accessed according to an ordering and then the content elements within the accessed page sequence elements are added to the page objects. Nowhere does the cited Saito anywhere teach or suggest the claim requirement of including content elements in page sequence elements to determine how to add content elements to page objects.

Because the cited combination of art does not teach or suggest all the claim requirements, alone or in combination, Applicants submit that the rejection of claims 8, 22, and 36 should be withdrawn.

B.   Rejection Under 35 U.S.C. §103(a) Over Adler in View of Saito and Barry

    1. Claims 5, 7, 19, 21, 33, and 35

Claims 5, 7, 19, 21, 33, and 35 are patentable over the cited art because they depend from base claims 1, 15, and 29, which are patentable over the cited art for the reasons discussed above.   Moreover, the dependent claims provide additional grounds of patentability over the cited art for the following reasons.

Claims 5, 19 and 33 depend from claims 2, 16 and 30 and require that the page objects include formatting properties in a third presentation language, where according to the intervening claims the page objects also include content and formatting properties in a first presentation language and the result document is in a second presentation language.

.   The Examiner cited Barry as teaching an additional presentation language, a page description language – image bit-map. Although multiple presentation languages may be known, Applicants submit that the Examiner has not cited any part of the combination of references that teaches that a page object includes content in the first presentation language used in the source document and formatting properties in a third presentation language, such that the page objects are generated from a result document in a second presentation language.

Because the cited combination of art does not teach or suggest all the claim requirements, alone or in combination, Applicants submit that the rejection of claims 5 and 33 should be withdrawn.

As mentioned, after filing the Appeal and before filing the Appeal Brief, Applicants submitted an amendment to claim 19 to change the dependency to intervening claim 16. If the Examiner declines to enter the Amendment to claim 19, Applicants

submit that claim 19 is patentable over the cited art because it depends from claim 15,

which is patentable over the cited art for the reasons discussed above. Additionally, if the

Examiner declines to enter this Amendment, Applicants request the Board to have the

amendment entered on the grounds it places the claims in better condition for

consideration on Appeal because amended claim 19 will depend from intervening claim

16 as do corresponding claims 5 and 33, which depend from intervening claims 2 and 30

and because intervening claim 16 provides antecedent basis for the second presentation

language element.

Claims 7, 21, and 35 are patentable over the cited art because they depend from

claims 5, 19, and 33, which are patentable over the cited art for the reasons discussed

above.

C.   Rejection Under 35 U.S.C. §103(a) Over Adler in View
      of Saito, Barry, and Sall

      1.   Claims 6, 20, and 34

Claims 6, 20, and 34 are patentable over the cited art because they depend from

base claims 1, 15, and 29 and intervening claims 5, 19, and 33, which are patentable over

the cited art for the reasons discussed above, and because the additional requirements of

these claims in combination with the base and intervening claims provide further grounds

of patentability over the cited art.

VIII.   Conclusion

Each of the rejections set forth in the Final Office Action is improper and should

be reversed.

Respectfully submitted,

  /David Victor/

Reg. No. 39,867
Konrad Raynes & Victor LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, California 90212
Tel: 310-553-7977
Fax: 310-556-7984

IX.   Claims Appendix

1.   (Previously Presented) A method for processing a source document in a structured document format including elements providing source content to render, wherein the source content comprises code that is rasterized into output, comprising:

receiving the source document including source content in a presentation language;

receiving a layout data structure separate from the source document, providing formatting properties specifying a layout and format of the content output, wherein the layout data structure does not include source content;

processing the source document and the layout data structure to determine formatting properties, including page divisions, for the content in the source document;

generating multiple page objects, wherein each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page, wherein at least one page object has multiple content elements, and wherein the content elements include content to place on the pages; and

transmitting the page objects to a rasterizer to transform into renderable information capable of being generated by an output device.

2.   (Previously Presented) The method of claim 1, wherein the presentation language comprises a first presentation language, further comprising:

transforming the source document and source content therein into a result document in a second presentation language, wherein the result document includes the source content and the formatting properties provided by the layout data structure,

wherein the formatting properties indicate page divisions of the content, and wherein the multiple page objects are generated from the result document.

3.   (Canceled)

4.   (Previously Presented) The method of claim 2, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), and wherein the page layout data structure comprises an XSL stylesheet.

5.   (Original) The method of claim 2, wherein the page objects include formatting properties in a third presentation language.

6.   (Original) The method of claim 5, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), wherein the third presentation language comprises the Mixed Object Document Content Architecture (MO:DCA), and wherein the layout data structure comprises an XSL stylesheet.

7.   (Original) The method of claim 5, wherein the third presentation language comprises a page description language.

8.   (Original) The method of claim 2, wherein the page objects include content and formatting properties in the second presentation language.

9.   (Original) The method of claim 2, wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO).

10.   (Original) The method of claim 1, wherein the page objects include content and formatting properties in a device independent presentation language.

11.   (Previously Presented) The method of claim 1, wherein the presentation language comprises a page description language.

12.   (Canceled)

13.   (Canceled)

14.   (Original) The method of claim 1, wherein the source document does not indicate page divisions for the content.

15.   (Previously Presented) A system for processing a source document in a structured document format including elements providing source content to render, wherein the source content comprises code that is rasterized into output, comprising:

an output device;

means for receiving the source document including source content in a presentation language;

means for receiving a layout data structure separate from the source document, providing formatting properties specifying a layout and format of the content output, wherein the layout data structure does not include source content;

means for processing the source document and the layout data structure to determine formatting properties, including page divisions, for the content in the source document;

means for generating multiple page objects, wherein each page object includes the source content in the presentation language used in the source document and the determined formatting properties for one page, wherein at least one page object has multiple content elements, and wherein the content elements include content to place on the pages; and

means for transmitting the page objects to a rasterizer to transform into renderable information capable of being generated by the output device.

16.   (Previously Presented) The system of claim 15, wherein the presentation language comprises a first presentation language, further comprising:

means for transforming the source document and source content therein into a result document in a second presentation language, wherein the result document includes the source content and the formatting properties provided by the layout data structure, wherein the formatting properties indicate page divisions of the content, and wherein the multiple page objects are generated from the result document.

17.   (Canceled)

18.   (Previously Presented) The system of claim 15, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), and wherein the page layout data structure comprises an XSL stylesheet.

19.   (Previously Presented) The system of claim 16, wherein the page objects include formatting properties in a third presentation language.

20.   (Original) The system of claim 19, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), wherein the third presentation language comprises the Mixed Object Document Content Architecture (MO:DCA), and wherein the layout data structure comprises an XSL stylesheet.

21.   (Original) The system of claim 19, wherein the third presentation language comprises a page description language.

22.   (Original) The system of claim 16, wherein the page objects include content and formatting properties in the second presentation language.

23.   (Original) The system of claim 16, wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO).

24.   (Original) The system of claim 15, wherein the page objects include content and formatting properties in a device independent presentation language.

25.   (Previously Presented) The system of claim 15, wherein the presentation language comprises a page description language.

26.   (Canceled)

27.   (Canceled)

28.   (Original) The system of claim 15, wherein the source document does not indicate page divisions for the content.

29.   (Previously Presented) An article of manufacture, in communication with an output device, for processing a source document in a structured document format including elements providing source content to render, wherein the source content

comprises code that is rasterized into output, and wherein the article of manufacture

comprises code capable of causing a processor to perform:

receiving the source document including source content in a presentation

language;

receiving a layout data structure separate from the source document, providing

formatting properties specifying a layout and format of the content output, wherein the

layout data structure does not include source content;

processing the source document and the layout data structure to determine

formatting properties, including page divisions, for the content in the source document;

generating multiple page objects, wherein each page object includes the source

content in the presentation language used in the source document and the determined

formatting properties for one page, wherein at least one page object has multiple content

elements, and wherein the content elements include content to place on the pages; and

transmitting the page objects to a rasterizer to transform into renderable

information capable of being generated by the output device.

30.   (Previously Presented) The article of manufacture of claim 29, wherein the

presentation language comprises a first presentation language, and wherein the code is

further capable of causing the processor to perform:

transforming the source document and source content therein into a result

document in a second presentation language, wherein the result document includes the

source content and the formatting properties provided by the layout data structure,

wherein the formatting properties indicate page divisions of the content, and wherein the multiple page objects are generated from the result document.

31.   (Canceled)

32.   (Previously Presented) The article of manufacture of 30, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), and wherein the page layout data structure comprises an XSL stylesheet.

33.   (Previously Presented) The article of manufacture of 30, wherein the page objects include formatting properties in a third presentation language.

34.   (Previously Presented) The article of manufacture of 33, wherein the first presentation language comprises the Extensible Markup Language (XML), wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO), wherein the third presentation language comprises the Mixed Object Document Content Architecture (MO:DCA), and wherein the layout data structure comprises an XSL stylesheet.

35.   (Previously Presented) The article of manufacture of claim 33, wherein the third presentation language comprises a page description language.

36.   (Previously Presented) The article of manufacture of claim 30, wherein the page objects include content and formatting properties in the second presentation language.

37.   (Previously Presented) The article of manufacture of claim 30, wherein the second presentation language comprises Extensible Stylesheet Language Formatting Objects (XSL-FO).

38.   (Previously Presented) The article of manufacture of claim 30, wherein the page objects include content and formatting properties in a device independent presentation language.

39.   (Previously Presented) The article of manufacture of claim 29, wherein the presentation language comprises a page description language.

40.   (Canceled)

41.   (Canceled)

42.   (Previously Presented) The article of manufacture of claim 29, wherein the source document does not indicate page divisions for the content.

43.   (Previously Presented) The method of claim 1, wherein generating page objects comprises:

adding at least one content element to one page object until the page object does not have available space for an additional content element;

adding at least one additional content element to at least one additional page object until all content elements are included in page objects.

44.   (Previously Presented) The method of claim 43, wherein page sequence elements include content elements, further comprising:

accessing page sequence elements according to an ordering of the page sequence elements, wherein the content elements within the accessed page sequence elements are added to page objects.

45.   (Previously Presented) The system of claim 15, wherein generating page objects comprises:

adding at least one content element to one page object until the page object does not have available space for an additional content element;

adding at least one additional content element to at least one additional page object until all content elements are included in page objects.

46.   (Previously Presented) The system of claim 45, wherein page sequence elements include content elements, further comprising:

accessing page sequence elements according to an ordering of the page sequence elements, wherein the content elements within the accessed page sequence elements are added to page objects.

47.   (Previously Presented) The article of manufacture of claim 29, wherein generating page objects comprises:

adding at least one content element to one page object until the page object does not have available space for an additional content element;

adding at least one additional content element to at least one additional page object until all content elements are included in page objects.

48.   (Previously Presented) The article of manufacture of claim 47, wherein page sequence elements include content elements, wherein the code is further capable of causing the processor to perform:

accessing page sequence elements according to an ordering of the page sequence elements, wherein the content elements within the accessed page sequence elements are added to page objects.

X. <u>Evidence Appendix</u>

None

XI.   Related Proceedings Appendix

     None